ending up on a system. However, I'm not sure I follow how this explains increase in *inbound* attacks on these systems.

It would be nice to see a "step back" section in the paper. For example, given you experience, is signature-based misuse detection the way to go? Would there be point in actively maintaining the signature set instead of accepting the crud that Snort's default ruleset with Emerging Threats spews into your logs?

## Response from the Authors

We made several changes to the paper to address the comments of the reviewers. Most reviewers noted that the description of the heuristic in Section 3.3 was not very clear. A major change was that we substantially clarified Section 3.3.

Answers to specific questions (in the beginning we mark the reviewer number):
R1: Our heuristic builds on attacks detectable by snort, which provides a rather rich set of signatures for detecting attacks of various types.
R1, R2: We clarified that our heuristic does not classify the type of a detected infection; it solely detects an infection.
R2: We made more clear what it automated. The detection process of our heuristic is entirely automated. It uses an assignment of snort rules into three classes, which classification we derived manually.
R2: Trivially combining alerts from a host and passing them to an administrator would not work because it results in a prohibitively large number of suspected systems an administrator needs to manually inspect.
R3: Using a set of detection parameters is common in most detection studies. We have used security tickets for re-mediated incidents and have physically visited the owners of infected systems to derive a reliable ground truth for fine-tuning the detection threshold of our heuristic. In addition, we extended Section 4.4 to report how the number of false positives/negatives changes with the detection threshold.
R3: We discuss in detail the root-causes of the main types of false positives in Section 4.3.
R4: We provide a reference for the 99% false-positives figure in the introduction. In addition, we confirmed this figure with our data too based on the suggestion of reviewer 5.
R4: We extended Section 4.4 to report how the false-positive/negative rate changes for different thresholds.
R5: Although alert correlation has been studied extensively in the past, compared to previous studies the main novelty in our work is the characterization of 9 thousand infected hosts in a production network.
R5: The authors agree that the false positive rate could be further reduced. Some hints for improving it are given in the discussion of the root-causes of false positives in Section 4.3. This could be interesting future work.
R5: We liked the suggestion of reversing the bundling process to compute the initial false positive rate. We followed this suggestion and found 99.4% false-positive un-aggregated alerts (excluding policy alerts). This confirms our expectation.